# CHINESE INFORMATION EXTRACTION AND RETRIEVAL

*Sean Boisen, Michael Crystal, Erik Peterson, and Ralph Weischedel*
BBN Corporation
70 Fawcett Street
Cambridge, MA 02138
weischedel@bbn.com
617-873-3496

*John Broglio, Jamie Callan, Bruce Croft*
University of Massachusetts
Amherst, MA

*Theresa Hand, Thomas Keenan, Mary Ellen Okurowski*
Department of Defense
Fort Meade, MD

## 0. ABSTRACT

This paper provides a summary of the following topics:

1. what was learned from porting the INQUERY information retrieval engine and the INFINDER term finder to Chinese
2. experiments at the University of Massachusetts evaluating INQUERY performance on Chinese newswire (Xinhua),
3. what was learned from porting selected components of PLUM to Chinese
4. experiments evaluating the POST part of speech tagger and named entity recognition on Chinese.
5. program issues in technology development.

## 1. BACKGROUND

As a reinvention laboratory, the TIPSTER Program offers the Government not only an opportunity to foster large scale research and development, but also avenues to deploy the resulting enhanced technologies. The primary focus of TIPSTER Phase One was to advance the state-of-the-art in document detection and information extraction through multiple contract awards for different algorithmic approaches. Large scale text collections were tackled by the detection contractors, while domain and language portability were the challenge for the extraction contractors. The Text Retrieval Conference (TREC) and the Message Understanding Conferences (MUC) evaluated and baselined the technology developments. In contrast, TIPSTER Phase Two focused on creating an architecture to integrate the two technologies, and on deploying these technologies at multiple Government agencies. The deployments were called demonstration systems because their success in daily use would demonstrate the capabilities of the technologies to end-users.

As a demonstration system, the goal was to port Phase One technologies to Chinese. The University of Massachusetts ported their INQUERY system with the development of HanQuery. BBN ported many of the major components of PLUM to Chinese and created Named Entity Identification capabilities. This paper describes program and technical issues identified during the joint Government-contractor effort and shares lessons learned in these two areas.

## 2. TECHNOLOGY ISSUES

### 2.1 Building a Chinese Retrieval System

### 2.1.1 General Issues

Information retrieval in a foreign language requires modification to text and user interfaces. Stemming, word boundary identification, punctuation and stopword identification must all be modified; appropriate input and presentation methods must be provided. But once these interface issues are resolved, the retrieval model and enhancement techniques operate equally effectively in all the languages we have worked with.

Text and user interface issues:

- Writing style varies according to language, including right-left, left-right, or top to bottom starting on the right.

| Report Documentation Page | *Form Approved*<br>*OMB No. 0704-0188* |
|---|---|

| | | |
|---|---|---|
| 1. REPORT DATE<br>**MAY 1996** | 2. REPORT TYPE | 3. DATES COVERED<br>**00-00-1996 to 00-00-1996** |

| | |
|---|---|
| 4. TITLE AND SUBTITLE<br>**Chinese Information Extraction and Retrieval** | 5a. CONTRACT NUMBER |
| | 5b. GRANT NUMBER |
| | 5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | 5d. PROJECT NUMBER |
| | 5e. TASK NUMBER |
| | 5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>**BBN Corporation,70 Fawcett Street,Cambridge,MA,02138** | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSOR/MONITOR’S ACRONYM(S) |
| | 11. SPONSOR/MONITOR’S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT

**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES

**TIPSTER TEXT PROGRAM PHASE II: Proceedings of a Workshop held at Vienna, Virginia, May 6-8, 1996. Sponsored by the Defense Advanced Research Projects Agency.**

14. ABSTRACT

**This paper provides a summary of the following topics I. what was learned from porting the INQUERY information retrieval engine and the INFINDER term finder to Chinese 2. experiments at the University of Massachusetts evaluating INQUERY performance on Chinese newswire (Xinhua) 3. what was learned from porting selected components of PLUM to Chinese 4. experiments evaluating the POST part of speech tagger and named entity recognition on Chinese. 5. program issues in technology development.**

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT<br>**unclassified** | b. ABSTRACT<br>**unclassified** | c. THIS PAGE<br>**unclassified** | **Same as Report (SAR)** | **11** | |

- The fundamental concept of what is an indexable word (or *term*) changes from language to language, as does the concept of a word stem or root. Some languages, like Chinese and Japanese, are written continuously, with no spaces between words.

  In Chinese, *artificial intelligence* becomes a four character phrase, which could be translated literally as *man-made-cognition-able*. How many words do we have here: one, two or three?

  In Semitic languages, words are classically viewed as consonant stems with the addition of prefixes and suffixes. The stems undergo changes in vowels or doubling of consonants. Finding a term stem for indexing can generate a lot of false relations between words.

  In agglutinative languages, a single printed word can express the lexical semantics of a complex noun phrase or even a whole sentence.

Character encoding:

- A given language may have multiple non-ASCII character encodings. Occasionally, as in SJIS, JIS and EUC for Japanese, there is a comparatively simple algorithmic mapping from one encoding to another. More typically, as in Chinese, different conversion tables use different character order and are not in one-to-one relation with each other.

Expectations:

- Users have different expectations. For example, although research indicates that a bigram model for languages like Chinese may be very effective, a user may be disconcerted to see the second character of one word juxtaposed with the first character of the following word as a search item.

### 2.1.2 Character Encodings

The ASCII encoding was evolved and standardized on the English language, so input and display of any other language presents problems for ASCII-oriented display technology and languages such as C, where even the datatype *char* is ambiguous and not guaranteed to support more than 7-bit ASCII. Because of this, many foreign languages have alternative character encodings. Languages that do not use the Roman alphabet may have any number of competing encodings in use by different agencies or in different countries or on different platforms.

Modern Chinese has two graphic character sets: PRC (*simplified*) characters and Taiwan (*traditional*) characters. (Classical Chinese has additional graphic character styles.). In the UNIX world, these two display sets are encoded by the GB (GuoBiao - PRC) and Big5 (Taiwan) two-byte eight-bit encodings, although on other systems (e.g., DOS) there are two-byte seven-bit encodings of these display character sets.

One problem that must be handled is a query in one character set retrieving documents encoded in different character sets. The query must be transcoded for retrieval from each database and the documents retrieved must then be transcoded into the input set for display. There will be information loss due to incomplete conversion tables or due to local expressions that have no equivalent in the another writing.

### 2.1.3 Indexing and Segmentation.

Our experience with both Japanese and Chinese has shown that character-based indexing is the most flexible approach to take for Chinese. Indexing each Chinese character as a term ensures that no information is lost.

Although it is possible to segment the documents into words automatically and index each word as a term, this can cause well-posed queries to fail for two reasons:

- Words can be improperly joined by the automatic segmentation.
- There are different understandings of the definition of a word. For example, the Chinese expression for *Beijing Institute of Physics* may be legitimately represented in a Chinese lexicon as a single word and a Chinese-speaking user may also perceive it as a word. But if this expression is stored as a single term, then perfectly reasonable queries such as *Physics institutes in China* or *Beijing technical institutes* would fail to match that term. For the same reason, query-time segmentation should include the raw characters or at least the bigrams in the query.

When the document index is character-based, then query-processing can determine proximity constraints based on word and phrase formation. A user may hand-segment the query, the query may be segmented automatically or adjacent bigrams from the query may be used.

Automatic segmentation in Chinese raises the special problem of name recognition. Foreign names are represented phonetically in Chinese by a small set of Chinese characters. These characters may appear individually in Chinese words, but when they are combined to sound out non-Chinese names they form sequences that are not otherwise part of a Chinese lexicon.

Chinese names present a different problem. There is a relatively small number of traditional Chinese surnames, but given names are essentially unrestricted combinations of two-character sequences. A Chinese name recognizer must look for sequences of unsegmented (or poorly segmented) characters, and try to identify a traditional family name, followed by two characters that could be a given name (i.e., not otherwise segmentable as a word or part of a word.)

Ideally name recognition should be efficiently interleaved with segmentation, so that when segmentation fails on a short sequence, the name recognizers can be called. A makeshift substitute for this interaction is to run the name segmenter to identify guaranteed names (from name lexicon), run the segmenter, and then run the name recognizer again, this time to identify possible names from the still unsegmented characters.

### 2.1.4 Query Processing

There are several issues in query processing besides those encountered by the user interface.

- The input character representation must be matched to the document collection representation, and converted if necessary.
- Characters which carry no meaning, such as punctuation or grammatical particles, should be discarded.
- Groupings of characters that represent words should be identified, either manually or automatically. This may include the special problem of Chinese and foreign name recognition.
- Query expansion methods. We relied on an automatic collection-driven concept-relation technology called InFinder described below and in [Jing].

In our first version of the Chinese IR system, we convert between whatever character sets are represented in our document database and whichever encoding the user has requested. We relied on user hand-segmentation to identify words. Our second version of the system has an automatic segmentation component. Where the user has indicated a preferred segmentation, however, it will be respected by the automatic component.

A future modification will be to combine the segmented query with the raw-character query, and possibly to break long words into their bigram subcomponents.

Query Expansion with Related Terms

One of the objectives of information retrieval with respect to the user is to render the technology more accessible by diminishing the gap between the retrieval performance of an expert or trained user and that of a novice or casual user. The InFinder technology shows a lot of promise in this area. The goal was to offer automatic or user-guided query expansion by supplying terms which are related in meaning to the query terms.

In the past, this has been attempted with a general-purpose thesaurus or with a keyword list or topic navigation outline. The general purpose thesaurus fails by bringing in terms which are unrelated to the usage or the context at hand, and by neglecting other terms which are germane to a query term in context. The topic navigation and keyword lists are very expensive to construct and fail in heterogeneous collections or in domains which change rapidly.

The InFinder technology constructs an automatic related-term database which attacks the two problems of currency relevance with the same mechanism. An automatic catalog is constructed from a collection based on word co-occurrence. Taking any word or phrase as a *concept*, the InFinder program collects and filters frequency information on the words that are most frequently found within two or three sentences of the concept of interest. Since all the information is gathered from the text collection at hand, the term relations are relevant to the text. The resulting database is an INQUERY database which can be updated as desired, so that as new usages appear in the text, they can be added automatically to the InFinder database.

When a query is submitted to the InFinder database for expansion, concepts which are contextually related to the query terms will be retrieved. Some number of the top terms can be automatically added to the original query to add coverage and specificity, or the user can be prompted to select which terms to add to the original query. In the user-guided approach, the user gets the added benefit of immediate feedback as to which concepts in the collection are related to the query. This information can lead to selection of a different collection, or modification of the original query to alter a term that has a domain-specific meaning not intended by the user. For the demonstration system, user-guided expansion was supported.

### 2.1.5 Relevance Feedback

In relevance feedback, selected documents are processed by the system, and terms which are suggested by those documents are added to the original query. Since the Chinese indexing is character-based, the relevance feedback approach treated characters as query enhancement terms. Since this did not produce good results, we modified the feedback selection techniques to select significant pairs of adjacent characters from the relevant documents (bigram model). This model appears to produce very good results, although the terms added are occasionally disconcerting for the user, since they represent parts of words, or characters from two different words that commonly appear together in a phrase.

We could segment the relevant documents so that we can use actual words in the feedback query. This will produce a more "readable" query, but ongoing research suggests that the results may be the same or worse than those produced by the bigram model. It is possible that a combination of bigram treatment with segmentation would produce consistently good results.

### 2.1.6 User Interface

To enable query input to the Chinese language version of INQUERY, it was desirable to have a graphical user interface platform that would allow the input and display of Chinese characters. While there is a great deal of grassroots support in the UNIX world for display of Chinese and Japanese (*kterm, cxterm*), documentation and stability are unreliable and they do not support sophisticated pointer-driven or menu-based interaction. The best candidate for a platform for a user interface was the New Mexico State University Computing Research Laboratory XAT library of *widgets* based on the Motif library for the X Window System. The XAT library supports display of several different languages, and two important characters encodings for Chinese: the *traditional* or *Big5* encoding, and the *simplified* or *GuoBiao* (GB) encoding. In addition, for both character sets, the XAT library supports several different input methods for both character sets, including both PRC and Cantonese *pinyin* and the *Standard Telegraphic Code* (STC) 4-digit numeric representation.

The XAT library would allow input of Chinese text, which could then be communicated to a program. It permits the program to display Chinese text by including an opening and closing annotation which indicated which character-encoding the text was using. It was often the case that collections were in the simplified character set, while the client users might be more familiar with the STC input method and/or the traditional character encoding and display. Therefore it was necessary to have the XAT library receive STC or Big5 encodings and display traditional characters, and to have INQUERY translate the traditional encodings into simplified characters to retrieve documents from a text collection. For this purpose, we used conversion programs provided freely on the network (GB-BIG5) or created at CIIR (STC).

## 2.2 Evaluation of the Prototype System

### 2.2.1 Evaluation Methodology

The purpose of evaluation is to assess retrieval effectiveness against some standards of expected performance. For information retrieval evaluations, a reasonably large set of documents is collected, a set of queries is prepared by domain experts, or collected from users, and the relevance of each document to each query is judged. In practice, the thoroughness of relevance judgments will vary. Only an extremely small collection of documents can be judged completely. For reasonably large sets, a subset of documents is identified and judged for each query. Then the performance of a system can be evaluated based on the subset of judged documents. This is an expensive and time-consuming procedure when done properly, requiring many months of work assembling queries and judging retrieved documents by domain experts.

A given system's performance will be reported in terms of *recall* and *precision*: recall indicates what percentage of all the relevant documents were retrieved at a given point; precision indicates what percentage of the documents retrieved were relevant. As recall increases to 100%, precision will decrease correspondingly.

The INQUERY technology has been formally evaluated in TIPSTER and TREC trials in English, Spanish and Japanese with outstanding results and comparable performance in each language. Since there is as yet no TREC track for a complete evaluation of Chinese IR systems, we have conducted an in-house evaluation with limited resources to determine if the quality of retrieval appeared to be in line with our performance in other languages.

We assembled thirty "natural language" queries, modeled on a current set of TREC queries, a typical query being: "Investment prospects in China for American companies". For each query we had a Chinese language expert examine and judge the ten documents ranked most highly by Chinese INQUERY. The queries were submitted in three different experimental sets: raw characters and two sets of word-based queries: hand segmented and automatically segmented.

The database used was the Chinese *Peoples Daily* collection containing more than 100 megabytes of text.

A second stage of the experiment tested relevance feedback on the same queries. Relevant documents were selected and two-character sequences common to the relevant documents were automatically added to the original query. The modified query was resubmitted to the system and the first ten documents returned were evaluated for relevance.

### 2.2.2 Evaluation Results

As the precision figures for the thirty queries in Table 1 show, even the unsegmented character-based queries give respectable results. On the average six out of the first ten documents will be relevant to a given query. Interpreted another way, the first document listed will be relevant in eight queries out of ten.

Hand segmentation requires the user to insert spaces between the Chinese words when entering the text of the query. As the table shows, this gives an average improvement in performance of about 10% over the unsegmented query. Automatic segmentation gives a similar increase in performance. The difference between the two segmentation methods is largely due the presence of proper names in the queries. Although we have developed a Chinese and foreign name recognizer, it was not used in the segmentation for this experiment. As a result names were interpreted as a series of characters.

The relevance feedback stage of the experiment was based on a bigram model, which means that a number of two-character sequences from the relevant documents were selected for query expansion. We have previously observed that two-character sequences perform much better than single-character selection in relevance feedback. It would also be possible to automatically segment the relevant documents for feedback analysis, but it is not clear that this method would produce a measurable difference within the parameters of this experiment.

As the table shows, relevance feedback gives a performance increase of 10-20%. Relevance feedback expands the original query, so the difference observed in the feedback experiment are due to the influence of the original segmented or unsegmented query terms.

### 2.2.3 Evaluation Conclusions.

Within the limitation of the evaluation methods, we can conclude that the performance of Chinese INQUERY is quite satisfactory and conforms to that of INQUERY in other languages.

Based on work in English and Japanese, it is expected that a combination method, combining a word-based query with its character-based raw text, would perform best. Based on the quality of our bigram-based relevance feedback, we also intend to experiment with a bigram method of segmentation. This would be faster and simpler than lexicon-based segmentation.. If used in a combination query, it is possible that the results would equal or surpass the more expensive automatic segmentation performance.

## 2.3 Extraction

### 2.3.1 Porting Components of the PLUM Information Extraction System to Chinese

The PLUM architecture is presented in Figure 1. Ovals represent declarative knowledge bases; rectangles represent processing modules. Gray elements are not yet available for Chinese. A more detailed description of the language-independent system components, their individual outputs (with examples for English), and their knowledge bases is presented in BBN's paper to the Sixth Message Understanding Conference (MUC-6). The processing modules are briefly described below.

**Message Reader.** The input to the PLUM system is the text of a document from the document manager, i.e., a "message". The message reader module determines message boundaries, identifies the message header information, and determines paragraph and sentence boundaries.

**Morphological Analyzer.** The first phase of processing is the Chinese segmenter developed and supported by New Mexico State University. The sequences of words found by the segmenter for each sentence is then assigned a part of speech, e.g., proper noun, verb, adjective, etc. In BBN's part-of-speech tagger POST, a bi-gram probability model and frequency models for known words (derived from large corpora) are employed to assign a part of speech to all words of the sentence in context.

**Lexical Pattern Matcher.** The Lexical Pattern Matcher was developed in 1992 to deal with grammatical forms, such as names in English and Japanese. It applies finite state patterns to the input, which consists of word tokens with part-of-speech. In particular, word groups that are important to the domain and that may be detectable with only local syntactic analysis can be treated here. For NE, named organizations, named persons, dates and times, monetary amounts, and percentages are found here. When a pattern is matched, a semantic form is assigned by the pattern.

The set of recognized entities is used by the output functions to SGML-mark the input.

**Fast Partial Parser (FPP).** The ultimate information extraction system for Chinese would include a grammar. No Chinese grammar is yet available for PLUM.

The FPP is a near-deterministic parser which generates one or more non-overlapping parse fragments

**Table 1**

| Top N | Raw Query | | | With Relevance Feedback | | |
|---|---|---|---|---|---|---|
| Docs | No_Seg | Hand_Seg | Auto_Seg | No_Seg | Hand_Seg | Auto_Seg |
| 1 | 83.3 | 93.3(12.05%) | 90.0 (8.09%) | 96.7 (16.13%) | 100 (20.09%) | 100 (20.09%) |
| 2 | 75.0 | 83.3 (11.11%) | 81.7 (8.98%) | 96.7(28.98%) | 93.3(24.44%) | 91.7(22.31%) |
| 3 | 76.7 | 78.9 ( 2.91%) | 76.7 (0.00%) | 87.8 (14.52%) | 90.0(17.39%) | 88.9 (15.95%) |
| 4 | 72.5 | 76.7 ( 5.84%) | 75.0 (3.49%) | 83.3 (14.94%) | 87.5(20.73%) | 85.8 (18.39%) |
| 5 | 69.3 | 74.0 ( 6.83%) | 74.0 (6.83%) | 78.0 (12.6%) | 82.7 (19.38%) | 82.7 (19.38%) |
| 6 | 66.7 | 71.1 ( 6.64%) | 72.2 (8.29%) | 74.4 (11.59%) | 80.0 (19.99%) | 78.9 (18.34%) |
| 7 | 64.3 | 68.9 ( 7.2%) | 70.0 (8.91%) | 69.5 ( 8.13%) | 78.6 (22.28%) | 76.2 (18.55%) |
| 8 | 60.8 | 66.3 ( 9.09%) | 68.8 (13.2%) | 67.5 (11.06%) | 75.4 (24.06%) | 74.2 (22.08%) |
| 9 | 59.2 | 64.8 ( 9.5%) | 65.9(11.36%) | 65.5 (10.69%) | 73.7 (24.54%) | 73.3 (23.86%) |
| 10 | 57.0 | 62.0 ( 8.82%) | 63.0 (10.57%) | 64.0 (12.33%) | 71.3 (25.13%) | 70.0 (22.85%) |

spanning the input sentence, deferring any difficult decisions on attachment ambiguities. When cases of permanent, predictable ambiguity arise, the parser finishes the analysis of the current phrase and begins the analysis of a new phrase. Therefore, the entities mentioned and some relations between them are processed in every sentence, whether syntactically ill-formed, complex, novel, or straightforward. Furthermore, this parsing is done using essentially domain-independent syntactic information.

**Semantic Interpreter.** Since no grammar is included, no semantic interpretation rules were written.

The semantic interpreter contains two sub-components: a rule-based fragment interpreter and a pattern-based sentence interpreter. The rule-based fragment interpreter applies semantic rules to each fragment produced by FPP in a bottom-up, compositional fashion. Semantic rules are matched based on general syntactic patterns, using wildcards and similar mechanisms to provide robustness. A semantic rule creates a semantic representation of the phrase stored with the syntactic parse.

**Discourse Processing.** Even without a grammar, semantic entities and relationships are still recognized and created by the lexical pattern matcher. These semantic representation are the input to the discourse component.

PLUM's discourse component creates a meaning for the whole message from the meaning of each sentence. The message level representation is a list of discourse domain objects (DDOs) for the top-level events of interest in the message (e.g., SUCCESSION events in the MUC-6 domain). The semantic representation of a phrase in the text only includes information contained nearby; the discourse module must infer other long-distance or indirect relations not explicitly found earlier and resolve any references in the text.

The discourse component creates two primary structures: a discourse predicate database and the DDOs. The database contains all the predicates mentioned in the semantic representation of the message. Any other inferences are also added to the database.

To create the DDOs, the discourse component processes each semantic form produced by the interpreter, adding its information to the database. The discourse component then applies inference rules that may add more semantic information to the discourse predicate database. When a semantic form for an event of interest is encountered, a DDO is generated and any slots already found by the interpreter are filled in. The discourse processor then tries to merge the new DDO with a previous DDO, in order to account for the possibility that the new DDO might be a repeated reference to an earlier one.

Once all the semantic forms have been processed, heuristic rules are applied to fill any empty slots from the text surrounding the forms that triggered a given DDO. Each filler found in the text is assigned a confidence score based on distance from trigger. Fillers found nearby are of high confidence, while those farther away receive worse scores (low numbers represent high confidence; high numbers low confidence; thus 0 is the "highest" confidence score).

**Template Generation.** For named entities, SGML is inserted into a copy of the message text.

For full template output, the output generator takes the DDOs produced by discourse processing and fills out the application-specific templates. Clearly, much of this process is governed by the specific requirements of the application, considerations which have little to do with linguistic processing. The template generator must address any arbitrary constraints, as well as deal with the basic details of formatting.

The template generator uses a combination of data-driven and expectation-driven strategies. First the DDOs found by the discourse module are used to produce template objects. Next, the slots in those objects are filled using information in the DDO, the discourse predicate database, other sources of information such as the message header (e.g., document number), or from heuristics (e.g., in MUC-6 terms, the type of an organization object is most likely to be COMPANY).

### 2.3.2 Porting Named Entity Extraction to Chinese

**Impact of segmentation.** One of the major challenges for Chinese named-entity extraction is the lack of explicit word boundaries in Chinese text. For a word-based named entity system like the one used by the TIPSTER demonstration system, this necessitates the use of a segmenter to preprocess the text. This dependence means that segmenter errors will greatly lower extraction accuracy. Unfortunately, the class of words most difficult to segment correctly are proper nouns such as person names and locations. Furthermore, a segmentation for a given text that is considered correct by one set of criteria may not be the segmentation most useful for named entity extraction.

Looking forward, an interesting project would be to combine the segmentation and extraction steps into one process, since many of the tasks of a segmenter (e.g. parsing out names of people and locations) dovetail nicely with named entity extraction.

**Rules for aliases.** Just as English abbreviations and aliases for named entities are formed by selecting letters or subsets of words from the phrase making up the entity name, Chinese aliases are also formed by selecting one of more characters from the entity. For locations this is generally just the first character of the location name. Aliases for person names are also fairly straight forward. For organizations, the alias is generally formed by selecting a character from each word

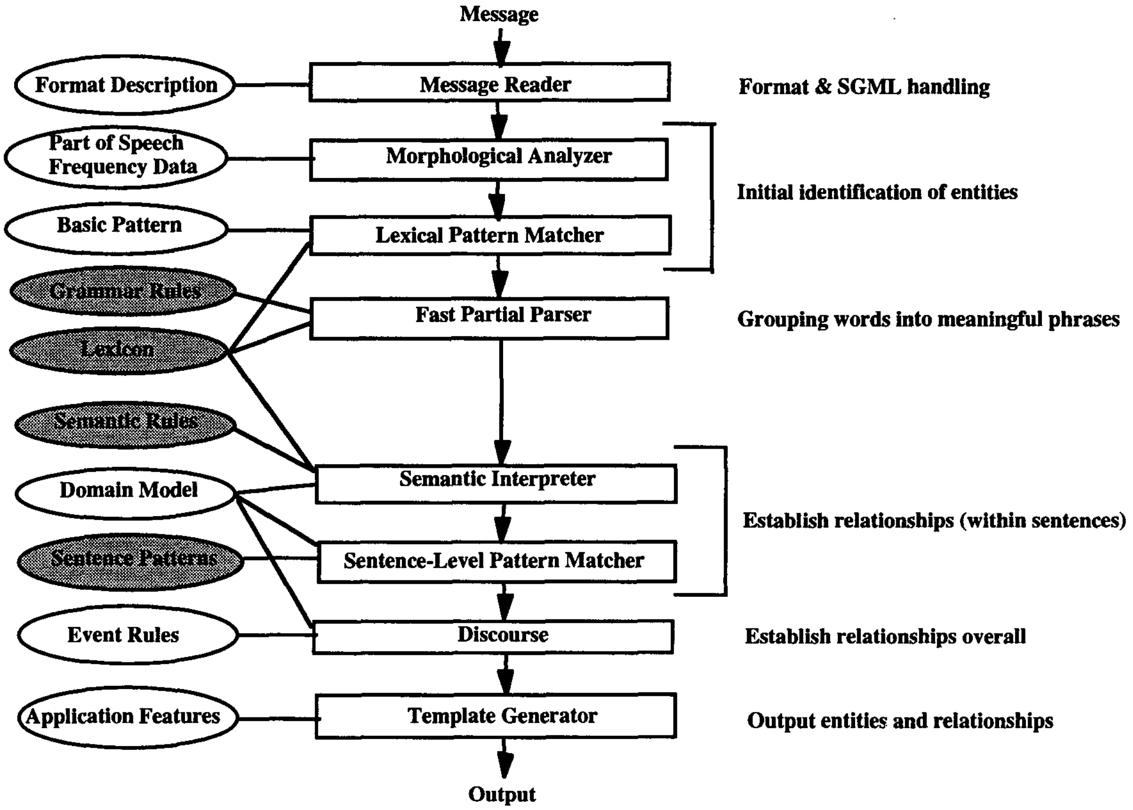

**Figure 1: Ultimate Chinese PLUM System:** Rectangles represent domain-independent, language-independent algorithms; ovals represent knowledge bases. Gray elements are not yet available for Chinese.

of the full organization name. However, the characters picked can and often do occur anywhere in the word, and no easy algorithm exists to determine which characters these are.

### 2.3.3 Assessment of the Merit of the Technical Approach and Lessons Learned

Prior to this effort, the PLUM information extraction system had been applied to several domains in English and to two domains in Japanese. Though that was varied experience, it was still limited experience with a high risk, high payoff technology.

#### 2.3.3.1 Difficulties Posed by Chinese

Chinese appears to be much harder than many other languages where information extraction has been attempted. Almost all data detection algorithms (for document storage and retrieval) and information extraction algorithms are word-based, i.e., they assume words for higher level processing. In many written languages, word boundaries are clearly marked by spaces. Chinese and Japanese, on the other hand, have no explicit indication of word boundaries; a reader must determine the writer's intended sequence of words, a process called word segmentation. Chinese segmentation seems inherently harder than Japanese, based on our experience. For instance, in Japanese, any change from Kanji characters to kana or to Romaji reliably signals a word boundary. Chinese has one uniform character set, and therefore does not provide as many easy boundaries. As a second example, consider foreign names. In Japanese, foreign names are typically transliterated into a sequence of easily identified kana characters, making recognition of foreign names rather easier than in Chinese, where foreign names are transliterated into the same character set as those used for common words.

The current PLUM architecture for Chinese separates segmentation, part-of-speech analysis, and name extraction into 3 pipelined modules. There are ambiguities in segmentation that probably can't be resolved without including more context in the decision. Combining these three modules into a single integrated model might improve performance, since similar information is used in all three decisions.

A second technical challenge in Chinese is in recognizing native Chinese first names. These are frequently common words with no capitalization to indicate whether the word is being used as a name or not.

A third problem is the lack of non-proprietary resources for Chinese. This suggests the need to develop resources such as lists of word plus part of speech, grammars, lexicons with syntactic features and at least high level semantic categories (person, organization, product, event, state of affairs, etc.).

#### 2.3.3.2 Lack of Linguistic Resources

One of the unexpected costs in this effort arose from the lack of linguistic resources. In our experience with Japanese, both a grammar and a list of roughly 35,000 words with their parts of speech were available. A list of words with their parts of speech is invaluable to having at least minimal syntactic/semantic information about words; it is presumed by any grammar. A grammar makes higher level processing possible. Developing either from scratch is quite time consuming.

A consequence of this was that more data labeled by part of speech (and word segmentation) was needed to process newswire than in any previous language we have worked on (English, German, Japanese, and Spanish). See the table below. Three factors seem critical: the amount of part of speech training data, whether the written language supports "ending analysis", and the size of the list of words plus their parts of speech. Note that the error rate can be well under 10% if either the spelling of the language supports ending analysis or there is a sizable list of words and their parts of speech (e.g., a dictionary listing part of speech for each entry). Neither was available in Chinese, where the error rate has been much worse than in any previous language we have worked on.

Whereas a corpus of 80,000 words marked in context by part of speech was adequate to give less than a 10% error rate in Japanese, in Chinese, with a corpus of 100,000 words marked, the error rate on newswire was still well over 10%, predominantly due to the fact that the error rate on unknown words in newswire was near 50%.

The high error rate on unknown words in Chinese is consistent with our experience with English; if ending

| Language | Part of Speech Training | Ending Analysis | Size of Word + PoS List | Error Rate |
|---|---|---|---|---|
| Japanese | 80,000 | No | > 35,000 | 3-4% |
| Spanish | 50,000 | Yes | 0 | 8% |
| English | 4,000,000 | Yes | 40,000 | 3-4% |
| Chinese | 100,000 | No | 0 | 12-15% |

**Figure 2:** Factors determining part-of-speech error rate.

analysis and capitalization are not employed the error rate on unknown words is roughly 50%. (By "ending analysis," we mean evidence of a word's part of speech given its spelling, e.g., the probability that a word is a noun, given it ends in "tion".)

Consider Spanish, which has a small phonetic alphabet. typical endings representing syllables can provide additional evidence as to the part of speech of an unknown word. With a corpus of roughly 60,00 words marked by part of speech, the overall error rate on newswire was below 10% (even though without a large list of words plus parts of speech), and the error rate on unknown words was only half that of Chinese.

Since neither capitalization nor ending analysis are available in Chinese, the only alternative to reducing the error rate in Chinese newswire is reducing the number of unknown words, e.g., by developing a list of words plus parts of speech.

In addition to the need for additional linguistic resources, clearer guidelines for developing these resources are needed. For example, the granularity of segmentation and the part-of-speech tag set must be appropriate for the applications and capabilities of the system modules that require them. For the demonstration system, part-of-speech data which was prepared early in the project, before all the requirements for downstream modules were clear, often had to be revised. Better software tools and procedures to support quality control are also needed, given the inherent difficulties in manually tagging large amounts of data.

### 2.3.3.3 Lessons Learned

We believe the following can be learned from this effort:

1. *Basic linguistic resources.* Given our assessment of the difficulty of processing Chinese, this suggests the need for development of basic resources for non-European languages, e.g., segmenters, word + part of speech lists, lexicons, and grammars.
2. *Linguistic expertise.* Personnel with linguistic expertise who are also programmers may be rare for some languages. In such cases, a development environment for non-programmers is highly desirable. Looking to the future, approaches to learning extraction rules from examples is research with very high payoff.
3. *System software.* System software to support languages other than English is still minimal, especially for languages not representable as ASCII characters, such as Chinese. As a result, underlying software, such as operating systems, programming languages, text editors, and user interfaces, require substantial effort for each new language; the associated costs to obtain them, install them, learn them, and work around their limitations are not going down.

## 3. PROGRAM ISSUES

Simply porting the components of TIPSTER advanced text processing technology is insufficient proof that a technology will actually perform as expected in a given language. Porting to a new language introduces an array of challenges.

### 3.1 Problem Definition

One way to reduce the risk of technology transfer is selecting a well-defined problem and scope it appropriately in the development and prototype stage. In the initial stages of development, it is tempting to select a problem that best matches the known technical capability of the systems. In order to create a useful system, however, the system implementor must work closely with future customers to identify a problem, while at the same time, bearing in mind that uncertainties in the technology extension process can complicate finding a match between an application problem and the technical capabilities. Even though the contract would have benefited from a joint Government and contractor requirements analysis, the central problem was not in understanding requirements but rather prototyping developing technologies.

The developer and system implementor must understand and agree on the risks involved in development, especially in the situation when advanced technology is being applied to a completely new domain or language. Are there sufficient resources available to support moving the technology to a new area? Is there language expertise available to interpret and explain the novel characteristics of the language? All of the involved parties must evaluate the severity of the risks on a successful system outcome. Positive experience in Phase One with Japanese led the Government and contractors to downplay the port to Chinese as a risk factor.

### 3.2 Evaluation of Capabilities

All involved parties should agree, in advance, on what constitutes a successful system development. If the components of the text technology successfully process foreign languages text, is that a sufficient test? Should the results of an empirical evaluation be similar to previous results in similar languages? Should rigorous evaluation metrics by employed? For the demonstration, the baseline evaluation metrics of MUC and TREC for Information Extraction and Information Retrieval, respectively, had not previously been applied to Chinese information technology. Text retrieval evaluation for Chinese will not be baselined until TREC-5, in 1997 and Chinese extractions results were not baselined until Spring 1996. Data preparation of topic descriptions for information retrieval and templates for information extraction is costly, but without defined evaluation data how is agreement reached on an acceptable level of performance. How do we manage expectations in an unknown situation? All must agree on the minimum accepted system performance to determine its success.

## 3.3 Software Integration

One of the key goals of the TIPSTER Phase II effort was to foster sharing of resources, including code reuse. The demonstration project was very ambitious in its support of this goal. The demonstration systems include software components developed under other contracts by New Mexico State University, including an early version of the TIPSTER Document Manager (TDM), a Chinese Segmenter, and a multi-lingual Motif text widget. The use of TDM was the primary means of demonstrating TIPSTER compliancy, another Phase II goal. Unfortunately, the original government time estimate for architecture definition was low, and a concrete definition of the architecture were not available during the demonstration design phase. Although one of the purposes of the demonstration systems was to provide valuable feedback in the iterative design cycle of the TIPSTER architecture development, this strategy, in retrospect, was detrimental to successful system development. Adherence to evolving architecture standards and commitment to reusing shared software impacted negatively on the demonstration systems. In addition, the shared software was immature, but the development schedules necessitated that it be robust.

## 3.4 Resource Identification

System planning necessitates identification and acquisition of essential resources, such as supporting data and software development tools. Developers must identify what types of resources are required for successful development, whether they are currently available or must be developed, and how soon in the development cycle they must be available. If the critical path of the system schedule depends on the timely acquisition or development of new resources, the schedule must allow for this. For many foreign languages, software tools are not readily available. This is especially true for languages which are not traditionally the focus of natural language or computer applications. The lack of availability of basic development tools, such as multi-lingual editors and fonts, can have a serious impact on development schedule. In order to minimize impact on the system deployment schedule, all required resources should be acquired prior to system development. Many delays were introduced into the effort by unavailability of infrastructure resources.

An additional resource issue is personnel management among multiple contract sites and the Government site. New combinations of technical expertise and create new opportunities from past contract experiences where all work is done by the contractor. How these resources are managed most effectively provides new challenges to both the Government and contract groups.

## 3.5 Schedule

The developer and the Government must devise a schedule that indicates approximate system delivery dates. These delivery dates should be adhered to, to the extent possible, and any slippages should be documented and the cause for the slippage understood. The developer should identify any dependencies in the schedule for system deployment. Developers need to manage "requirements creep" and identify potential negative impacts on scheduling initiatives. For technologies being ported to a new language, with a heavy dependence on creation of new resources, developers should track incremental progress to more readily identify problem areas and potential schedule slippage.

## 3.6 Support

Life cycle support of advanced natural language technology is still beyond the ability of most software centers, which creates an unrealistic support requirement on contractors that focus primarily on research and technology development. The support structure for such a system must be developed and refined as the technology matures in order to be able to handle any future problems.

While the goal of sharable systems across multiple Government agencies is admirable, all participating agencies must commit to providing support and infrastructure resources to maintain the resulting system within each office. Common system development will provide benefits to the Government in the long term, however it requires substantial initial investment and customer buy in.

## 3.7 Lessons Learned

The design and development of the demonstration system was a valuable learning experience, which will positively impact the success of future technology efforts. Among the most relevant lessons:

- Keep the scope of technology al development efforts small, until an advanced technology is proven to work for a given language.
- Do not rely on baseline evaluation results to predict the success of a technology effort in a new language. Anticipate that unforeseen challenges of a new language will probably drive system performance down to some degree.
- Determine reasonable, cost effective means for evaluating new capabilities in existing technologies.

- Do not rely on evolving standards as a core component of a system.

- Avoid depending on component software still under development without including support for coordination between main system and component developers.

- Include support for any component software as a separate task for project scheduling and budgeting purposes.

- Begin technology development only after the support infrastructure is identified. Develop effective management mechanisms for multiple site coordination with the Government.

- Track incremental progress during the course of system development. This will allow the system integrators and customers to more easily identify potential problem areas.

- Expect that systems shareable across Government agencies require interagency investment far beyond the initial definition of an architecture.

The TIPSTER demonstration system allowed us to test implementation of TIPSTER technology in a new language, and gave us a more complete understanding of risks involved in undertaking such an effort. Hopefully this increased understanding will benefit us in future TIPSTER advanced technology transfer efforts.

[Jing] Jing, Y.; Croft, W.B. (1994) An association thesaurus for information retrieval. *Proceedings of RIAO 94*, 146-160.